Discrete Pattern Recognition by Fitting onto a Continuous Function

ALIETTE COSSÉ-BARBI, MOURAD RAJI*

Institut de Topologie et de Dynamique des Systèmes (ITODYS), CNRS URA-34, Université Paris 7-Denis Diderot, 1 rue Guy de la Brosse, 75005 Paris, France

Received 16 June 1996; accepted 17 June 1997

ABSTRACT: This article outlines an original method for matching discrete structures when atom correspondences are unknown. This method avoids the current atom-by-atom treatment (and its inherent combinatorial problems) and considers the structures to be compared in their totality. The basic idea is to first obtain the atom correspondences by fitting one of the two discrete structures onto a spline approximation of the other, rather than optimizing in discrete space, and, second, to overlap the two discrete structures on the basis of the proposed assignment. As starting data, the method requires only the Cartesian coordinates of the two structures. No connectivity information, neither atom labeling nor matching tolerance is required. This method can readily handle matches of molecules with a few hundred atoms. It is able to search for a given 3D pattern as well as for a pattern common to two structures. © 1997 John Wiley & Sons, Inc. *J Comput Chem* **18**: 1875–1892, 1997

Introduction

The recognition and the analysis of three-dimensional (3D) similarities between molecules is of fundamental importance for the interpretation and prediction of their physical, chemical, or biological properties; that is, molecules with similar shape features are expected to interact in a similar way with radiation (circular dichroism), reagents (chemical reactivity), or biological receptors (bioreactivity).

*Presented as M. Raji's thesis at Université Paris 7-Denis Diderot

Correspondence to: A. Cossé-Barbi; e-mail: cosse@Paris7.jussieu.fr

For the understanding of molecular properties, both overall shape features and local shape features may be relevant. Indeed, the shapes to be compared can be overall ground states, delocalized three-dimensional arrangements of not necessarily connected functional groups (pharmacophores[1]), or local arrangements around a functional group (chromophores, reactive sites).

The molecular shape can be a discrete nuclear arrangement or a 3D molecular body (electron distribution or, at a simpler level, van der Waals volume).

Well-documented 3D data bases, experimentally (x-ray or neutron diffraction[2,3]) or computationally generated,[4] have provided a completely new perspective. Overcoming the viewpoint of

CCC 0192-8651 / 97 / 151875-18

quantitative structure activity[5] or $\rho\sigma$[6] relationships on particular series of compounds, they make it possible to compare many dissimilar molecules on purely 3D criteria, in order to design compounds belonging to new series and having interesting shape features.

Since 1973, following Gund's proposals,[7] computational programs well adapted to 3D data bases have been developed. Current methods for examining 3D similarities fall into two categories:

1. Comparison of areas or volumes (van der Waals volumes,[8–10] electron densities,[11,12] etc.) requiring a continuous optimization procedure.
2. Comparison of atom positions or of interatomic distances requiring an optimization in discrete space.[13–18]

Whatever the comparison, type (1) or (2), programs are able to overlap only small structures (a few tens of atoms). It is easy to match discrete structures if the atom correspondences are known. In this case, we have only to determine a translation and rotation step for one molecule to overlap it with the other.

In fact, the atom correspondences are rarely known. The challenge is precisely to find them. Some programs search for a predetermined 3D pattern (SubStructure search) in a structure. Others search for a three-dimensional pattern common to two or more structures (Common SubStructure search). Any method for handling the latter problem must be able to solve the former, but the reverse is not true. Whatever the aim (SS or CSS), it is necessary to find the atom correspondences before overlapping the structures.

This atom correspondence search leads to a combinatorial problem. The possible atomic correspondences to be screened increase dramatically with molecular size. Techniques such as exhaustive tree searches with branch-and-bound pruning,[19] the use of neural networks,[20] or simulated annealing[21] are designed to reduce the combinatorial problem.

Clearly, the difficulty is inherent in the discrete nature of the entities to be matched. We propose to overcome it by going from discrete to continuous space. The basic idea is as follows:

- First, we do not try to match the two discrete entities but simply to move one of the entities (the smaller) on a continuous representation of the other (the greater). This fitting leads to an atom assignment.
- Second, we control this assignment and adjust the atom positions of the two discrete entities more closely.

To prevent possible misunderstanding about the method it is useful to make some points immediately. As starting data, the method requires only Cartesian coordinates of the two entities to be compared. Atom labeling (although this could be introduced in further improvements), connectivities, and any kind of prescreening are unnecessary. Moreover, the atom assignments are sought by minimizing a function. Consequently, as opposed to current methods based on atom-by-atom treatment, the match is not constrained by any matching tolerance.

Method for Circumventing the Combinatorial Problem

Current methods for 3D discrete pattern recognition proceed in the following order:

- They establish that the discrete substructure is contained in the discrete structure.
- If this is the case and, if necessary, they compute the substructure translation and rotation step **T** required to overlap the SS on the structure.

We propose to proceed in the reverse order. We search for the rotation and translation step **T**′ whose existence allows us to assume the inclusion of the substructure in the structure. Consequently, the 3D subgraph search problem becomes a step-search analytical problem.

PRINCIPLE OF THE METHOD

Figure 1 summarizes the principle of the method in two-dimensional (2D) space. Let us suppose that the problem to solve is to find a discrete four-atom pattern (points ▲) included somewhere in a seven-atom structure (points ♦). Current methods have to screen 840 possible correspondences. We avoid this painstaking screening in the following way. The seven-atom structure is first interpolated by a continuous function called $\mathscr{R}$ (Fig. 1b) and the structure atoms are momentarily put aside (Fig. 1c). Let us call **T**′ the translation

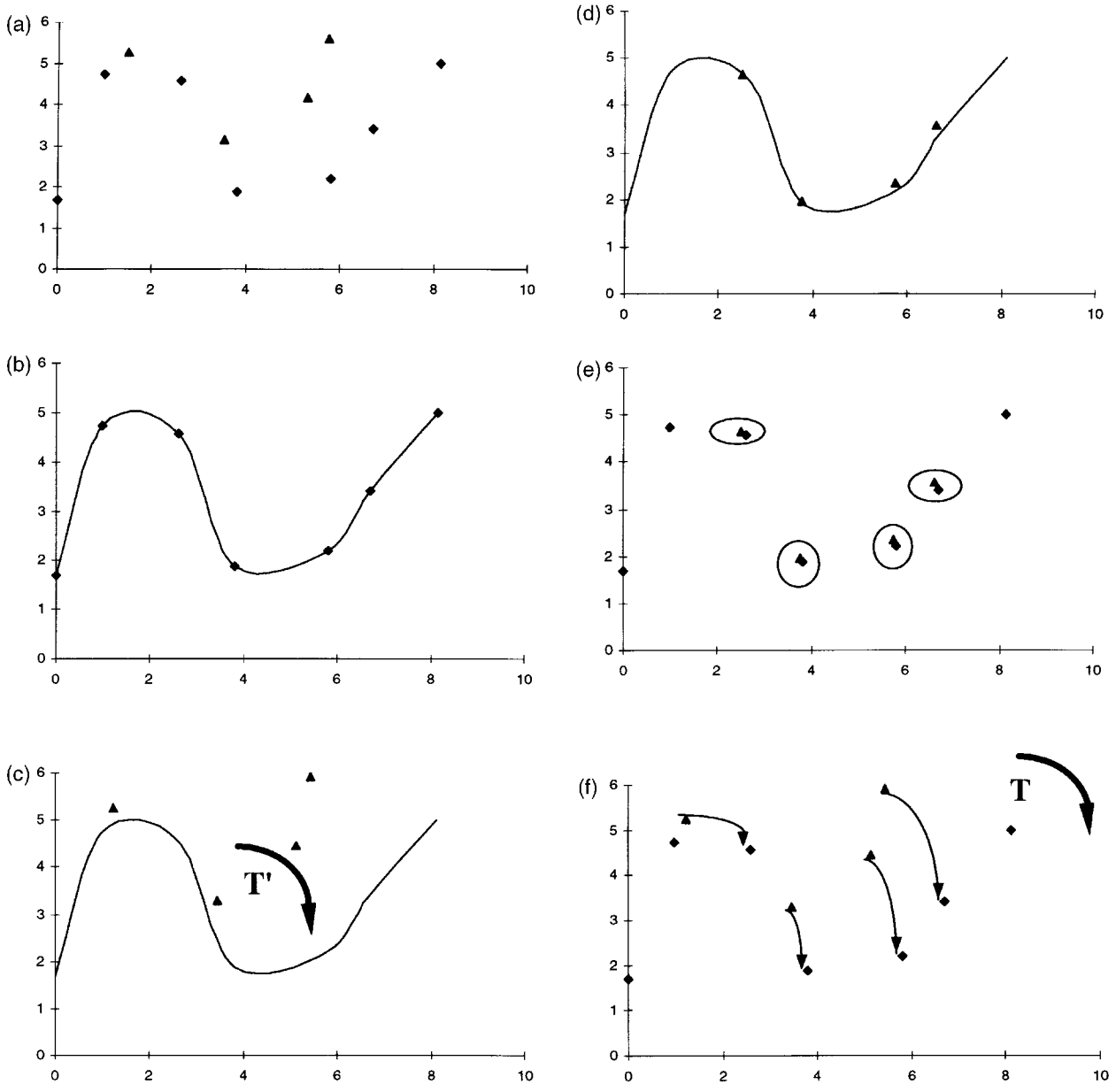


FIGURE 1. Principle of method. (a) Problem to solve. (b) Introducing continuous interpolation, $\mathscr{R}$, of the structure. (c) Setting aside structure atoms. (d) Fitting substructure pattern onto $\mathscr{R}$. (e) Deducing atom assignments. (f) Return to initial situation, assignments being known. (g) Fitting two discrete patterns.

and rotation step necessary to fit the four-atom pattern on $\mathscr{R}$.

Then, the four-atom pattern is fitted on $\mathscr{R}$ and $\mathbf{T}'$ is computed (Fig. 1d). At this stage, continuous interpolation is abandoned. We turn our attention back to the seven atoms of the structure. Each substructure atom is assigned to the nearest structure atom (Fig. 1e). With the atom assignment being known, we return to the initial situation (Fig. 1f). Let us call $\mathbf{T}$ the rotation and translation step necessary to fit the substructure pattern on the structure one. The substructure pattern is fitted on the structure pattern and $\mathbf{T}$ is computed (Fig. 1g).

GENERALIZATION IN 3D SPACE: DETAILED DESCRIPTION

Going from Discrete to Continuous Space

The continuous representation of the structure is obtained by projecting its *Ns* atoms onto two

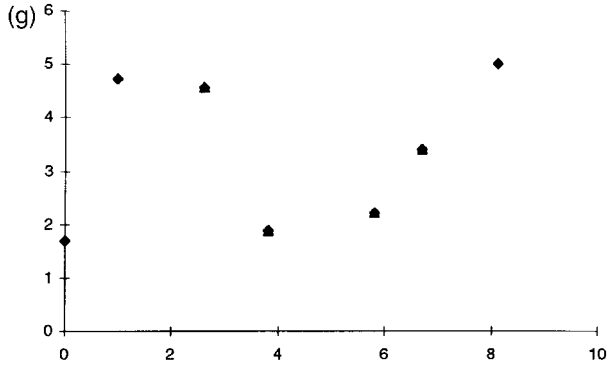


FIGURE 1. (Continued)

planes of a Cartesian coordinate system and interpolating through the projections.

Let us denote, by P_{xy} and P_{xz}, the interpolation functions in planes O_{xy} and O_{xz}, respectively. For each atom At_k of the structure with atom coordinates x_k, y_k, and z_k, we have:

$$\begin{aligned} P_{xy}(x_k) &= y_k \\ P_{xz}(x_k) &= z_k \end{aligned} \qquad (1)$$

This continuous representation, $\mathscr{R}$, is not unique and contains an extra item of information (all the points in between the real projection of the discrete structure), but it presents the only relevant property; that is, it contains all the geometrical information. This $\mathscr{R}$ representation is associated with the structure. It can be stored to constitute a structural data base, and used when necessary for particular 3D pattern searches.

The interpolation function may be any function provided that it is derivable and continuous at each projection point. A polynomial representation was attractive because polynomial parameters can be stored easily. We have discarded Legendre polynomials whose major drawback is that the degree of the polynomial increases with the structure size and have chosen a cubic spline interpolation.[22] Here, whatever the structure size, the polynomials have the same degree (three) and it is only the number of parameters that increases with size. However, some caution is necessary to ensure continuity and derivability at each point and we have to examine more closely: (i) the ends of the $\mathscr{R}$ representation; and (ii) situations in which two atoms have the same projection.

(i) The polynomial representation starts "before" ($x < x_k$ min) and finishes "after" ($x > x_k$ max) the set of projection points. This is obtained by adding two virtual points to the real projections in each plane xy and xz. These virtual points are chosen very far away (1000-Å variation in each coordinate) from the "first" (x_k min) and the "last" (x_k max) real projection points to assist the convergence process (**T′** search).

(ii) Two atoms might be projected in the same point. In other words, we could have two y or z values (or more) for the same x. In this case, a polynomial representation is nevertheless created by increasing one of the two y (or z) values by a small amount, 10^{-5} Å for instance. However, it is important to notice that the structure coordinates are not modified, the 10^{-5} Å increment being introduced only to make it possible to construct the continuous representation.

T′ Step-Search

T′ moves the projection of the substructure atoms, At_i, whose coordinates are x_i, y_i, and z_i onto the $\mathscr{R}$ representation. **T′** depends on six parameters, the three rotation angles, θ_x, θ_y, and θ_z, around the three axes, Ox, Oy, and Oz, and the three translations, tr_x, tr_y, and tr_z.

Depending on the order chosen for the individual translations and rotations, there are many steps to adjust the 3D discrete pattern on the $\mathscr{R}$ representation. Our purpose being not to find an optimal translation and rotation step, we adopt the following arbitrary order:

$$\mathbf{T}' = Ttr_z \circ Ttr_y \circ Ttr_x \circ T\theta_z \circ T\theta_y \circ T\theta_x \qquad (2)$$

T′ is written as eq. (3):

$$\mathbf{T}' = \begin{vmatrix} \cos(\theta_y) \times \cos(\theta_z) & \sin(\theta_x) \times \sin(\theta_y) \times \cos(\theta_z) & \cos(\theta_x) \times \sin(\theta_y) \times \cos(\theta_z) & tr_x \\ & -\cos(\theta_x) \times \sin(\theta_z) & +\sin(\theta_x) \times \sin(\theta_z) & \\ \cos(\theta_y) \times \sin(\theta_z) & \sin(\theta_x) \times \sin(\theta_y) \times \sin(\theta_z) & \cos(\theta_x) \times \sin(\theta_y) \times \cos(\theta_z) & tr_y \\ & +\cos(\theta_x) \times \cos(\theta_z) & -\sin(\theta_x) \times \cos(\theta_z) & \\ -\sin(\theta_z) & \sin(\theta_x) \times \cos(\theta_y) & \cos(\theta_x) \times \cos(\theta_y) & tr_z \\ 0 & 0 & 0 & 1 \end{vmatrix} \qquad (3)$$

and denoting by x'_i, y'_i and z'_i the At_i coordinates after $\mathbf{T}'$ transformation, we have:

$$\begin{aligned}
x'_i &= \cos(\theta_y) \times \cos(\theta_z) \times x_i \\
&\quad + \big(\sin(\theta_x) \times \sin(\theta_y) \times \cos(\theta_z) \\
&\quad - \cos(\theta_x) \times \sin(\theta_z)\big) \times y_i \\
&\quad + \big(\cos(\theta_x) \times \sin(\theta_y) \times \cos(\theta_z) \\
&\quad + \sin(\theta_x) \times \sin(\theta_z)\big) \times z_i + tr_x \\
y'_i &= \cos(\theta_y) \times \sin(\theta_z) \times x_i \\
&\quad + \big(\sin(\theta_x) \times \sin(\theta_y) \times \sin(\theta_z) \\
&\quad + \cos(\theta_x) \times \cos(\theta_z)\big) \times y_i \\
&\quad + \big(\cos(\theta_x) \times \sin(\theta_y) \times \sin(\theta_z) \\
&\quad - \sin(\theta_x) \times \cos(\theta_z)\big) \times z_i + tr_y \\
z'_i &= -\sin(\theta_y) \times x_i + \sin(\theta_x) \times \cos(\theta_y) \times y_i \\
&\quad + \cos(\theta_x) \times \cos(\theta_y) \times z_i + tr_z
\end{aligned} \tag{4}$$

If the 3D pattern is included in the structure with exactly the same spatial disposition of atoms, its projection after the translation and rotation step $\mathbf{T}'$ must be included exactly in the continuous $\mathscr{R}$ representation. We must have:

$$\begin{aligned}
P_{xy}(x'_i) &= y'_i \\
P_{xz}(x'_i) &= z'_i
\end{aligned} \tag{5}$$

In a realistic case, a 3D pattern is never found in a molecule with exactly the same spatial disposition of atoms. Therefore, the most we can do is to find the best fit of the substructure projections on $\mathscr{R}$. To achieve this, an obvious way is to minimize a quantity (or quantities) related to local distances parallel to the y and (or) z axis between the substructure atom projections and the polynomial representation (Fig. 2). One can use two different optimizations in the two projections planes, followed by the necessary interrelation between them, or minimize a quantity extended over all substructure atom projections, whatever the projection plane xy or xz. We have found this second way to be the more convenient. Therefore, we propose to search for the six parameters that minimize the following overall quantity, QT:

$$QT = \sum_{i=1}^{Nss} \big(P_{xy}(x'_i) - y'_i\big)^2 + (P_{xz}(x'_i) - z'_i)^2 \tag{6}$$

with Nss being the number of SS atoms.

At this stage, it is clear that computing QT requires only the substructure atom Cartesian coordinates and the parameters of the polynomial representation of the structure.

Solving eq. (6) requires only numerical methods. Many tools are possible. Here we use the BFGS algorithm,[23] because it converges, even in cases where we start far from the solution.

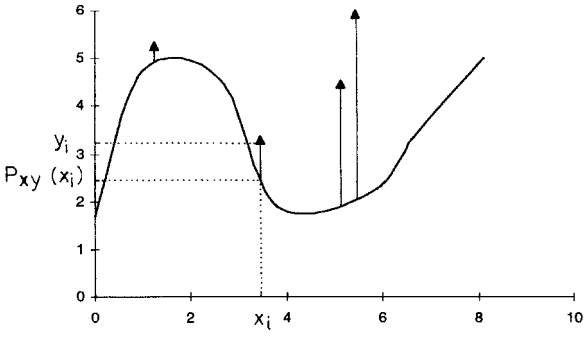


FIGURE 2. Local distances (vertical segments) involved in *QT*.

Atom Assignment

$\mathbf{T}'$ moves the substructure atom projections onto the $\mathscr{R}$ representation, *on* or *in between* the real atom projections of the structure. The proximity of the substructure and structure atom projections is the basis for atom assignment, each SS atom being assigned to the nearest structure atom by screening a distance matrix. This procedure could be a combinatorial problem in itself. To circumvent this drawback, we proceed in the following way. A first screening deals with all atoms pairs At_i, At_k with local distances $\|At_i\ At_k\|$ less than (or equal to) 0.05 Å. Two atoms, At_i and At_k, are paired if the local distance between them does not exceed 0.05 Å. These two atoms are then set aside. After this first screening, all local distances $\|At_i\ At_k\|$ to be considered exceed 0.05 Å. We then look for the minimum local distance, and the two atoms At_i and At_k corresponding to this lowest local distance are paired and set aside. This second screening is repeated until each substructure atom is matched with a structure atom.

This atom assignment step provides a unique correspondence for strong similarities between the two patterns as well as for strong dissimilarities. There are two reasons for this:

- First, for chemical patterns, it is impossible to find two substructure atoms At_i located at 0.05 Å (or less) from the same structure atom At_k or the same substructure atom located at 0.05 Å (or less) from two structure atoms.

- Second, it is very improbable to find two substructure atoms (or structure atoms) at the same distance from the same structure (substructure) atom. If this were the case, we would abandon the resulting combinatorial problem, because, by varying the SS initial location (see ''Repeating the Entire Procedure'' Section), we could recover the lost isomorphism if this latter were of interest.

T Step Determination: Inclusion Accuracy

The accuracy of the assignment must be controlled. For this purpose, we compute the rotation and translation step **T** by minimizing the root mean square distance between the corresponding atoms of the two discrete entities.

Let us call x_i'', y_i'', and z_i'' the Cartesian coordinates of the SS atoms after the **T** transformation; the *RMS* is written:

$$RMS = \sqrt{\frac{\sum_{1}^{Nss} (x_i'' - x_k)^2 + (y_i'' - y_k)^2 + (z_i'' - z_k)^2}{Nss}} \tag{7}$$

This *RMS* measures the accuracy of the assignment of the predetermined 3D pattern of atoms to the structure atoms. The isomorphism being known, the range of possible algorithms to determine **T** is larger than for **T**′. Nevertheless, as for **T**′ computation, we use the BFGS algorithm here also.

To summarize, one can see that the entire procedure just described does not require any local or overall threshold. For this reason, it always provides an atom assignment and the corresponding best match, whatever the similarity or the dissimilarity of the two discrete patterns to be compared. Consequently, in contrast to other methods based on narrow local adjustments and comparisons with local thresholds, *our method does not establish the absence of a match, but only gives its accuracy*. The user is left free to accept or to reject the match. His diagnostic is helped both by the overall criterion, the *RMS*, and by an ordered table of local distances $\| At_i \; At_k \|$.

REPEATING THE ENTIRE PROCEDURE

Why a Scan? QT Multiple Minima

The function, *QT*, always has several minima and the convergence process may lead to a local one while the absolute one is sought (i). In some applications, several minima are sought (polymeric structures (ii) and not necessarily the best ones (common pattern recognition, iii). Whatever the aim, it is necessary to repeat the entire procedure by modifying the initial location of the substructure with respect to the $\mathscr{R}$ representation.

(i) *Seeking the best overlap*. If the two patterns are identical (or almost identical), there is a unique way (or very few ways, perhaps two or three) to adjust the substructure on $\mathscr{R}$. Conversely, if the dissimilarity is strong and the adjustment poor, there are many ways of adjusting SS on $\mathscr{R}$. The first one found may lead to a nonoptimal assignment. In this case, better solutions must be sought.

(ii) *Patterns included many times in a structure*. Some organic and bioorganic materials are polymers reproducing the same pattern many times with small variations. Local 3D dissimilarities are of interest for detection in these polymeric structures because they are involved in the reactivity of such systems. For example, it is well known that the reactivity of ARN structures is related to minute variations in the 3D sugar pattern; the opening of DNA base pairs concomitant with the approach of a drug involves the kinking of the double strand and the interaction of a protein with a receptor involves folding. To recognize many similar patterns, a scan is necessary.

(iii) *Common pattern recognition*. Later we will extend the algorithm to the recognition of patterns common to two structures. In such a search for pharmacophoric patterns, the best assignment may not be relevant of particular biological or biochemical application and other assignments corresponding to poorer overlaps and local *QT* minima may be of greater interest. Moreover, the overall similarity criterion, the *RMS*, allows us to compare only common patterns of similar size. For two common patterns of different sizes, there is no means of determining which is the best. To recognize common patterns, our algorithm takes advantage of the multiple *QT* minima.

How to Scan

The most convenient way to seek other isomorphisms is to scan along the x axis. In this way, we take advantage of the particular form chosen for the $\mathscr{R}$ representation with the same variable, x_k, for the two spline interpolations, P_{xy} and P_{xz}. To make the scan efficient, we must: (i) prepare the structure; (ii) carefully define the alignment space; and (iii) choose a step for varying the initial location of the substructure pattern:

(i) Before the alignment procedure, the structure is rotated so that its largest extension is along the x axis.

(ii) To begin the scan, the substructure is translated along the x axis to make its greatest x_i coordinate equal to the smallest x_k coordinate of the structure. The scan ends when the smallest substructure x_i coordinate becomes greater than the greatest x_k coordinate of the structure.

(iii) The step along the x axis separating two SS initial locations must be chosen carefully. We have tested this point for structures in which a given pattern is repeated many times with some geometrical variation. If the step is too large, some occurrences will be missing and, if it is too small, the same solution will be obtained many times. In our applications, we have found 1 Å to be a reasonable value, but other applications could require different values.

Assessment of Algorithm Performances

The algorithm, including the numerical tools, the BFGS algorithm, and the spline approximation, is written in C language and run on an Alpha Server 2100. The aim of this section is to test its performances by comparing it with other methods.

HOW TO COMPARE OUR RESULTS WITH OTHERS

The literature provides timings for several algorithms. Nevertheless, it is impossible to compare them, because the range of hardware is too large and there is no simple way of putting CPU times on the same scale.

Moreover, some methods, such as simulated annealing, neural networks, and our method, minimize a function sometimes called an objective function. In contrast with atom-by-atom treatments, they do not use *any matching tolerance*. In these methods, the conditions for stopping the convergence process (iteration number, the smallest allowed difference between two successive values of the function to minimize, etc.) may depend on the tools chosen to ensure convergence (BFGS algorithm, etc.). For these reasons, one cannot reproduce a study similar to that of Brint and Willett comparing four methods by running them on the same computer with the same distance tolerance.[17] Nevertheless, we can try to answer several questions:

1. For substructures exactly included in a structure with the same disposition of atoms, or for substructures very slightly perturbed from the exact one, the exact assignment is known. Is the method able to produce the exact assignment?
2. What is its behavior when the dissimilarity between the two patterns is increased?
3. How does it behave when the sizes of the patterns to be matched increase?

It is not easy to isolate the role of structure size from that of substructure size. One can keep the latter constant and vary the former. In this case, of the total numbers of atoms, the fraction to be recognized (Nss/Ns) varies and this parameter may seriously affect the fit of the discrete pattern onto the continuous representation. One can also work with a fixed Nss/Ns ratio $(1, 0.5, \ldots)$ and change the structure size. To address these issues, we performed three studies.

First, we tested the ability of the algorithm to produce an exact match by using identical coordinates for the two discrete entities but a different numbering for the atoms and, of course, a different location in 3D space, with the substructure covering half the structure ($Ns = 2\,Nss$) or the entire structure ($Ns = Nss$). The molecular sizes were 20, 70, 134, and 316 atoms. For the first three ($Ns = 20$, 70, and 134), the Cartesian coordinates[24] were extracted from the Cambridge Crystallographic Database.[2] The last structure, with 316 atoms, was an oligonucleotide whose coordinates were obtained[25] by an empirical calculation (JUMNA program[26]) with NMR constraints. For the eight cases in Table I, the number of possible atom correspondences varies from $4.7 \cdot 10^{11}$ ($Ns = 2\,Nss = 20$) to $4.6 \cdot 10^{652}$ ($Ns = Nss = 316$).

TABLE I.
Search for a Given Pattern Exactly or Inexactly Included in a Structure. Comparison with Two Methods based on Minimization of an Objective Function.[a]

Method	Computer	Sizes	SS perturbation percentage	Correct assignment if available	Final mean distances (Å)	CPU times (seconds)
Neural networks[20]	DS 5000 / 200 DEC	*Ns* = 26 *Nss* = 5	0 [± 3%]	20–82% 15–55%		1.2 1.8
Simulated annealing[21]	IBM 3084 Q	*Ns* = *Nss* = 20 *Ns* = *Nss* = 70 ×7.5 *Ns* = *Nss* = 150	0 0 0	86–100% 73–92% 45–82%	0–0.095 0.09–0.028 0.016–0.082	0.18–1.68 10–51 ×(> 430) 132–1490
		Ns = *Nss* = 20 *Ns* = *Nss* = 150 ×7.5	±16% ±16%	— —	— —	0.53–4.88 157– > 1080 ×(> 140)
Discrete versus continuous (this work)	AS 2100	*Ns* = *Nss* = 20 *Ns* = *Nss* = 70 ×15.8 *Ns* = *Nss* = 134 *Ns* = *Nss* = 316	0 0 0 0	100% 100% 100% 100%	0 0 0 0	0.65 0.94 ×5.8 2.31 3.78
		Ns = 20 *Nss* = 10 *Ns* = 70 *Nss* = 35 ×15.8 *Ns* = 134 *Nss* = 67 *Ns* = 316 *Nss* = 158	0 0 0 0	100% 100% 100% 100%	0 0 0 0	0.55 0.79 ×5 1.28 2.76
		Ns = *Nss* = 20 70 ×6.7 134	±8% ±8% ±8%	— — —	0.43 0.53 0.44	1.16 1.69 ×3.9 4.56
		Ns = *Nss* = 20 70 ×6.7 134	±20% ±20% ±20%	— — —	0.94 1.01 0.99	1.18 3.90 ×5.6 6.63

[a]Patterns exactly included: mean times for ten different numberings and locations in space. Patterns inexactly included: mean times for ten substructures with different numberings and locations in space and differently perturbed from the exact one.

In a second test, in addition to the numbering and overall location variations, we introduced some random variation ($\pm 8\%$, $\pm 20\%$) in the positions of the pattern points.

In a third test, the *Nss/Ns* fraction to be recognized was systematically changed.

Table I compares our results with those of two methods based on the minimization of an objective function, the neural network method[20] and simulated annealing.[21] Table II compares our results with atom-by-atom treatments,[13–18] devoted especially to recognizing small substructure patterns.

ISOMORPHISM

In the case of an exact inclusion, and whatever the atom number (Table I), our method always gives the correct assignment with a zero *RMS*. This is not the case for neural networks or simulated annealing. For the former, the best solution is obtained for 20% to 82% of the runs and, for the latter, the number of points correctly assigned is better but the similarity criterion deduced from the difference distance matrix (DDM) is rarely zero by the end of the procedure.

The aim of our method is not to find all possible assignments. On the contrary, we wish to avoid most of them and, of course, the worse ones, and to detect the best ones as quickly as possible. However, in a study presented elsewhere on geometric chirality scales, we compared our method with a method proposed by Rassat,[27] based on the Hausdorff distance.[28] In the framework of this comparison, it was of interest to determine if all correspondences could be obtained. For the overlap of a dissymmetrical triangle with its ''enantiomer'' in 2D space, our method detects the three relevant isomorphisms of the six possible. The other three permute two SS atoms so that these atoms are not paired with the nearest structure

TABLE II.
Search for a Given Pattern Exactly Included in a Structure. Comparison with Atom-by-Atom Treatments.[a]

Method	Computer	Sizes	Matching Tolerance	Correct Assignment	CPU Times (seconds)
Lesk[17]	Prime 9950	*Ns* = 60 *Nss* = 5	0.25 Å	100%	1.36
		Ns = 60 *Nss* = 9	0.25 Å	100%	1.31
		Ns = 60 *Nss* = 15	0.25 Å	100%	4.54
Set reduction[17]	Prime 9950	*Ns* = 60 *Nss* = 5	0.25 Å	100%	0.80
		Ns = 60 *Nss* = 9	0.25 Å	100%	2.62
		Ns = 60 *Nss* = 15	0.25 Å	100%	8.96
Clique detection[17]	Prime 9950	*Ns* = 60 *Nss* = 5	0.25 Å	100%	1.21
		Ns = 60 *Nss* = 9	0.25 Å	100%	2.94
		Ns = 60 *Nss* = 15	0.25 Å	100%	9.03
Ullman[17]	Prime 9950	*Ns* = 60 *Nss* = 5	0.25 Å	100%	0.27
		Ns = 60 *Nss* = 9	0.25 Å	100%	0.43
		Ns = 60 *Nss* = 15	0.25 Å	100%	0.92
Discrete versus continuous (this work)	AS 2100	*Ns* = 70 *Nss* = 6	none	100%	0.82
		Ns = 70 *Nss* = 9	none	100%	0.78
		Ns = 70 *Nss* = 15	none	100%	0.68
		Ns = 70 *Nss* = 35	none	100%	0.79
		Ns = 70 *Nss* = 52	none	100%	0.99
		Ns = 70 *Nss* = 70	none	100%	0.94
		Ns = 316 *Nss* = 14	none	100%	2.94
		Ns = 316 *Nss* = 40	none	100%	2.88
		Ns = 316 *Nss* = 79	none	100%	2.03
		Ns = 316 *Nss* = 158	none	100%	2.76
		Ns = 316 *Nss* = 237	none	100%	3.89
		Ns = 316 *Nss* = 316	none	100%	3.78

[a] Mean times for ten different numberings and locations in space.

atoms. The same experiment on the overlap of a tetrahedron and its enantiomer leads to the 12 relevant isomorphisms of the 24 possible ones.

INCREASING STRUCTURE SIZES

Even if we cannot compare the absolute CPU times required by different methods, we can compare the time variation when the structure size is increased. For substructures exactly included in a structure (Table I), multiplying the structure size by about 16 multiplies the CPU time by five- or sixfold depending on Nss/Ns (0.5 or 1). This behavior must be appreciated by comparing it with the simulated annealing method in which the CPU time is multiplied by more than 430 when the structure atom number is multiplied by 7.5.

For substructures inexactly included in a structure, the times increase slightly, but the observation is similar. Multiplying the structure size by about seven multiplies the CPU time by about four- or sixfold, depending on the amount of substructure perturbation with respect to the exactly included one ($\pm 8\%$ or $\pm 20\%$). With the simulated annealing method the CPU time is multiplied by more than 140 when the structure size is increased by a factor of 7.5.

INCREASING SUBSTRUCTURE SIZES

Atom-by-atom treatments were applied to recognize small SS patterns up to a quarter of the structure. The time then increases monotonically with the substructure size. Our method behaves differently. The performances depend very slightly on Nss/Ns. Shallow minima are obtained for substructures covering about a quarter of the structure.

CPU TIME CONSIDERATIONS

The time variations for increasing pattern sizes are summarized in Figures 3 and 4. The time dependence is roughly logarithmic with respect to structure size (Fig. 3) whether the substructure covers half or all of the structure. It follows that our algorithm might be particularly attractive for structures with a few hundred atoms. With respect to SS size (Fig. 4), the time seems to depend on at least two conflicting factors: (i) Substructure covering most of the structure ($Nss/Ns = 1$) have very few ways to adjust on $\mathscr{R}$ while SS, corresponding to small Nss/Ns values, have many possible locations on $\mathscr{R}$. (ii) To assign atoms, we screen a distance matrix, whose size ($Ns \times Nss$) increases with the two pattern sizes.

Finally, increasing the SS size facilitates the fit on $\mathscr{R}$, but slows down the atom assignment step, while the structure size affects mainly the time required to assign the atoms. The dependence of the CPU time on structure size could be a consequence of the particular technical solutions chosen to simplify the assignment step.

We conclude that going from discrete to continuous space eliminates combinational problems and provides a fast and reliable method for determining pattern atom correspondences.

Further Adaptations and Applications

TAKING CERTAIN PROPERTIES INTO ACCOUNT

In the previous tests we fitted the discrete pattern onto the continuous $\mathscr{R}$ representation by minimizing a quantity QT [eq. (6)] roughly related to the ''distance'' between the discrete pattern and the $\mathscr{R}$ representation. Only the geometrical parameters were taken into account.

However, in some applications, the nature of atoms and/or their ability to bind with a receptor site are relevant properties for a 3D structure comparison. The quantity, QT, to be minimized can be modified to take into account this type of problem.

Let t_i be a numerical substructure atom property. QT could be, for example:

$$QT = \sum_{i=1}^{Nss} \left(P_{xy}(x'_i) - y'_i\right)^2 + (P_{xz}(x'_i) - z'_i)^2 + (P_{xt}(x'_i) - t'_i)^2 \quad (8)$$

Thus, t_i is the At_i atomic number if the user wishes to match only similar atoms. The property t_i may be defined differently if we wish to match all the atoms with lone pairs, whatever their nature. In this case, a zero t_i value is assigned to atoms with lone pairs and a value of $t_i = 1$ to the others.

In addition, the property t_i may be weighted by p_i in eq. (9):

$$QT = \sum_{i=1}^{Nss} \left(P_{xy}(x'_i) - y'_i\right)^2 + (P_{xz}(x'_i) - z'_i)^2 + p_i(P_{xt}(x'_i) - t'_i)^2 \quad (9)$$

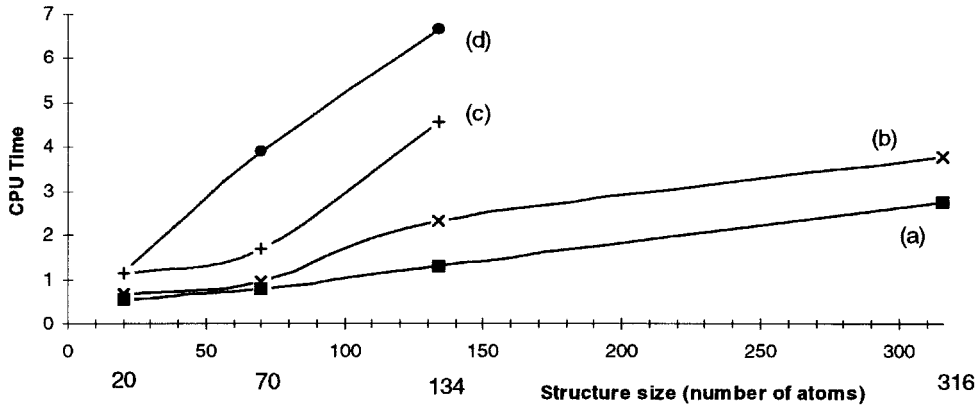


FIGURE 3. CPU (seconds) dependence on structure size (number of atoms). (a) Substructure exactly included and covering half the structure (*Ns* = 2*Nss*). (b) Substructure exactly included and covering all the structure (*Ns* = *Nss*). (c) Substructure randomly perturbed by ±8% from the exact one (*Ns* = *Nss*). (d) Substructure randomly perturbed by ±20% from the exact one (*Ns* = *Nss*).

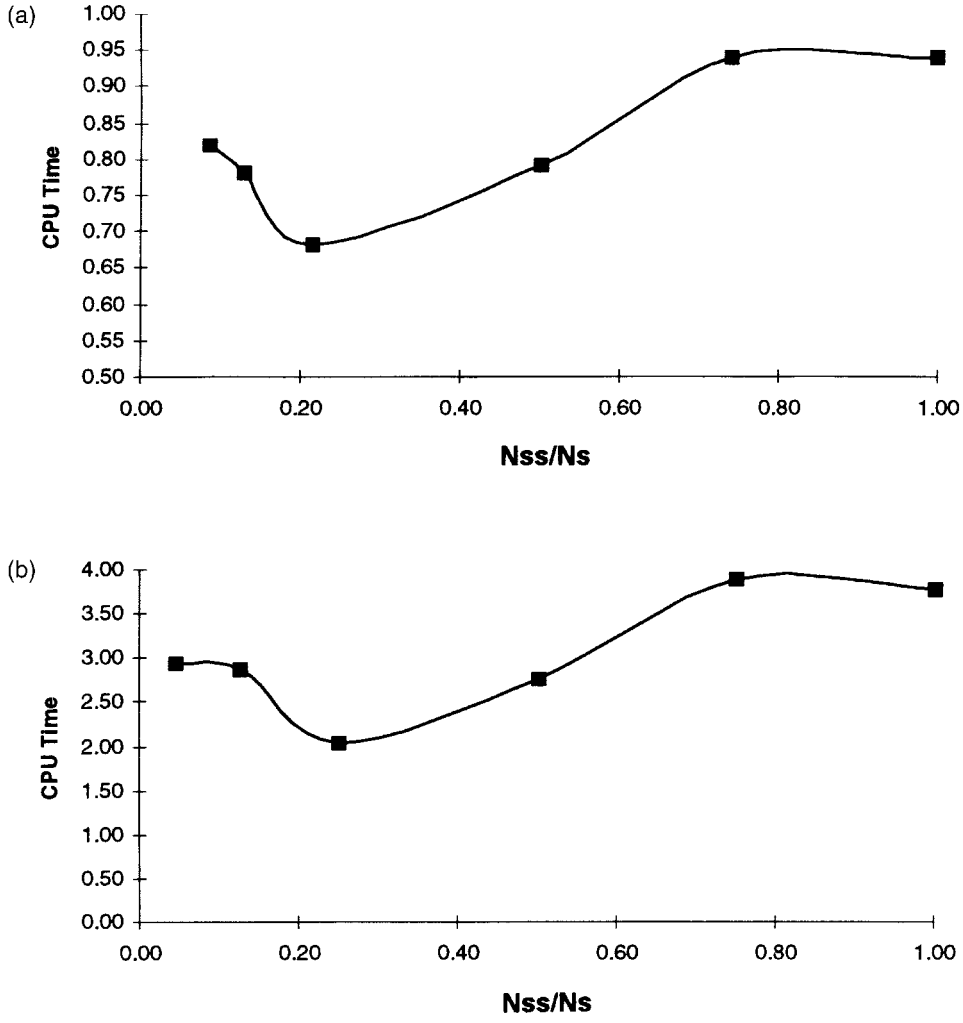
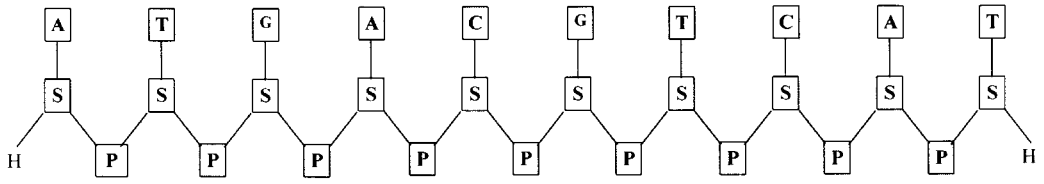


FIGURE 4. CPU (seconds) dependence on *Nss* / *Ns*. (a) *Ns* = 70. (b) *Ns* = 316.



SCHEME 1. P:Phosphodiester; S:Sugar; A:Adenine; T:Thymine; G:Guanine; C:Cytosine.

In more complex applications, some hydrogen bonding heteroatoms can sometimes act as acceptors and sometimes as donors. Consequently, we have to search not only for the matching of acceptors with acceptors and donors with donors but also for the matching of atoms bearing the two properties with donors or with acceptors. To take into account this possibility, we assign two numerical properties, t_{i1} and t_{i2}, to one atom:

$$t_{i1} = t_{i2} = 1 \quad \text{donor}$$
$$t_{i1} = t_{i2} = 2 \quad \text{acceptor}$$
$$t_{i1} = 1,\ t_{i2} = 2 \quad \text{donor and acceptor}$$
$$t_{i1} = t_{i2} = 4 \quad \text{neither of the properties}$$

QT to be minimized is modified as follows:

$$QT = \sum_{i=1}^{Nss} \left(P_{xy}(x'_i) - y'_i \right)^2 + \left(P_{xz}(x'_i) - z'_i \right)^2 + p_i \left(P_{xt1}(x'_i) - t'_{i1} \right)^2 \times \left(P_{xt2}(x'_i) - t'_{i2} \right)^2 \quad (10)$$

RECOGNITION OF A 3D PATTERN CONTAINED MANY TIMES WITH SOME STRUCTURAL VARIATION

A group from our laboratory (Dodin and Cordier) is presently working on the interaction of drugs with small oligonucleotides.[29] The binding site may be modified by some local irregularity.

The structure studied here (scheme 1) is a single strand with sequence ATGACGTCAT. It contains ten deoxyriboses (S = Sugar), nine phosphodiesters (P), and ten bases (three adenines, two guanines, two cytosines, and three thymines).

The first nucleotide unit has one sugar H atom more than the others and, for the last, the phosphate group is missing (Scheme 1). Borderline effects are expected for these two units.

Search for Third Nucleotide Unit (GSP)

Our aim is to recognize the third nucleotide, a 33-atom pattern including a guanidine base (G), a sugar, and a phosphodiester. Of the ten structure nucleotides, two of them bear a guanine base, the third and the sixth units, and the other eight differ from the third in the nature of the base. To fit the discrete pattern on the spline approximation of the strand, we take into account the atom nature by means of relationship shown in eq. (8).

Table III gives the best matches ordered according to their decreasing similarity with the third unit. As expected, units bearing a purine base, guanidine or adenine, are more similar to the third unit than those bearing a pyrimidine base (thymine, cytosine).

TABLE III.
Recognition of a 33-Atom Three-Dimensional Pattern, the Third GSP Nucleotide Unit, in the 316-Atom Structure, and Overlap With the Other Nucleotide Units.

Unit to be Recognized	Match with the:	Number of Atoms Assigned in the Unit	*RMS* (Å)
Third GSP	Third (GSP)	All	0.00
Third GSP	Sixth (GSP)	All	0.56
Third GSP	Ninth (ASP)	S,P atoms, +11 base atoms	1.05
Third GSP	Fourth (ASP)	S,P atoms, +12 base atoms	1.11
Third GSP	First (ASP)	S,P atoms, +9 base atoms	1.29
Third GSP	Second (TSP) etc.	S,P atoms, +7 base atoms	1.31

Our technique identifies the two (third and sixth) units bearing a guanine base, whereupon each atom of the substructure corresponds to an atom in the *n*th (third or sixth) unit.

Our technique also matches the GSP substructure with units differing from the third in the nature of the base. The accuracy then decreases and the match is only partial, with some atoms of the GSP substructure corresponding to atoms in the *n*th unit and others to atoms not in the *n*th unit. The *RMS* value and the number of atom correspondences in the *n*th unit allow us to appreciate the similarity between the third and the *n*th unit. The search could be continued by the determination of a maximal pattern common to the two units.

Search for the Third Deoxyribose

Our technique finds the third sugar and the nine others differing slightly from the third (Table IV) and each atom of the substructure corresponds to an atom in the *n*th sugar. If we exclude the first and last units, we have to note that odd units (fifth, seventh, ninth) are more similar to the third unit than even units (second, fourth, sixth, eighth).

An *a posteriori* examination of the sugar geometries shows that these odd and even sugars belong to different classes[30]: S with high phases and low amplitudes for the former, X with low phases and high amplitude for the latter.

SEARCH FOR COMMON SUBSTRUCTURES

Up to now, we have put off the introduction of any matching tolerance. However, if common patterns are to be retrieved, this can no longer be avoided.

TABLE IV.
Recognition of a 13-Atom Three-Dimensional Pattern, the Third Sugar, Contained 10 Times With Some Variation in the 316-Atom Structure.

Unit to Recognize	Match with the:	Number of Atoms Assigned in the Unit	*RMS* (Å)
Third S	First S	All	0.15
Third S	Second S	All	0.15
Third S	Third S	All	0.00
Third S	Fourth S	All	0.15
Third S	Fifth S	All	0.11
Third S	Sixth S	All	0.15
Third S	Seventh S	All	0.10
Third S	Eighth S	All	0.15
Third S	Ninth S	All	0.01
Third S	Tenth S	All	0.22

Technical Adjustments

Let us now recall our simplified example in 2D space (''Principle of the Method'' Section and Fig. 1) and suppose that at the end of the entire procedure, the situation is as follows. The four-atom pattern is fitted on the seven-atom pattern. Both the *RMS* distance and the local distances are computed.

Let us now suppose (Fig. 5) that the *user* considers that the distance between atom 4 of the substructure and its corresponding atom in the structure is too great. Atom 4 is put aside and the remaining three-atom pattern is fitted on the seven-atom one. The local requirement being achieved, we can conclude that we have found a common three-atom 3D pattern.

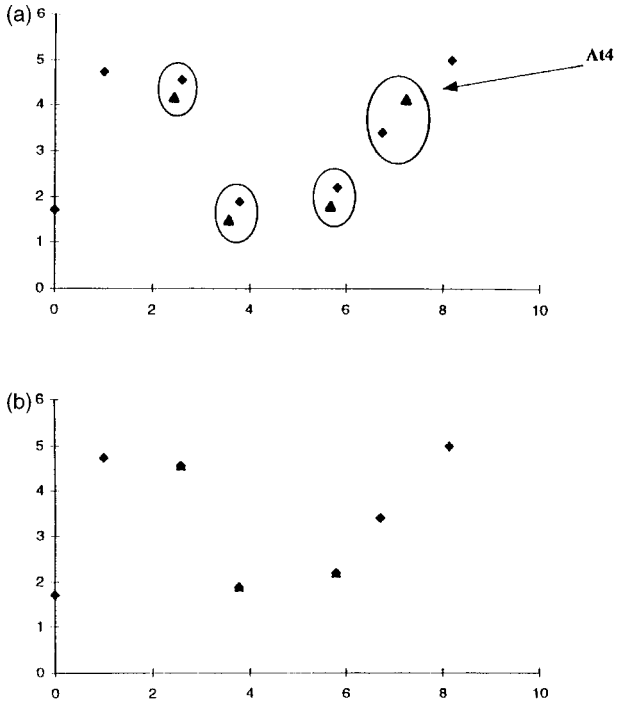


FIGURE 5. Common 3D patterns: some necessary technical adjustments. The smaller pattern (four atoms: ▲) is sought in the bigger (seven atoms: ♦) and after the full procedure. (a) the preliminary result is inspected: an SS atom, At_4 is considered to be too far away from its corresponding atom in the structure. (b) At_4 is set aside and the remaining three-atom pattern is fitted on the seven-atom one. The two patterns have this three-atom pattern in common.

More precisely, to search for common patterns in two structures, we proceed in the following way. The smaller structure plays the role of the substructure and we search for its inclusion in the larger structure. Here, the user has to make a decision: to define a local threshold, ξ. The local distances between the corresponding atoms are compared with ξ for each pair of atoms. If the local distances are smaller than ξ for each pair, the two structures are considered to have the smaller structure pattern in common. If the local distance between two or more corresponding atoms exceeds the local threshold, the substructure atoms concerned are deleted and we fit the remaining substructure 3D pattern on the structure. The local requirement is achieved. Thus, we have found a common 3D pattern, the size of which is given by the number of corresponding atom pairs, *Npairs*.

The *RMS*, limited to the corresponding atom pairs, measures the accuracy of the common substructure determination:

$$RMS = \sqrt{\frac{\sum_{i}^{Npairs} (x_i'' - x_k)^2 + (y_i'' - y_k)^2 + (z_i'' - z_k)^2}{Npairs}} \tag{11}$$

It can easily be seen that screening all substructure atoms whose local distances from the corresponding structure atom exceed ξ provides a unique solution, whatever the order of atom deletion. This solution depends only on the initially proposed isomorphism (see ''Atom Assignment'' section) and on the chosen local threshold.

Application to Substructure Common to Two Marine Neurotoxins

Saxitoxin (STX) and tetrodotoxin (TTX) are two marine neurotoxins selected by Danziger and Dean[19] and by Feuilleaubois et al.[20] to test their matching techniques. These toxins act on nerve endings by binding the sodium-channel macromolecule.

Their 3D structures are not obviously similar. For toxins, they are exceptionally small and their crystallographic structures, limited to the 22 (TTX) and 21 (STX) heavy atoms, are known.

Danziger and Dean searched for correspondences only between heteroatoms with one property, donor or acceptor, in common. They used a tree search technique and pruned the tree to limit the search. The objective function to be minimized is the *RMS* value of the distance difference matrix (Δd). Table V shows the proposed matches, with the atom numbering being the same as in the article by Dean and Chau[31] (Fig. 6).

Feuilleaubois et al. looked for a nine-atom 3D pattern of STX in TTX, taking into account the

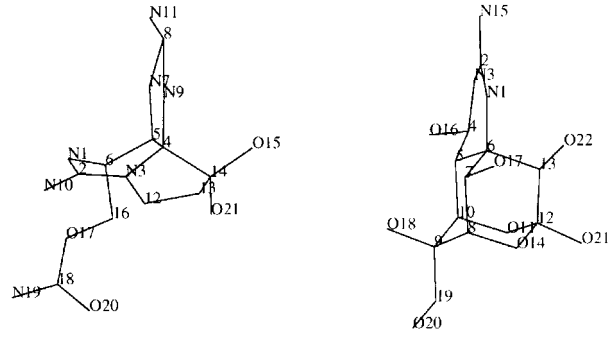


FIGURE 6. Saxitoxin (left) and tetrodotoxin (right). Numbering from Dean and Chau.[31]

TABLE V.
Tetrodotoxin (TTX) and Saxitoxin (STX) Common Substructures (CSS) Proposed by Danziger and Dean.[19]

STX	CSS Size	D N7	D N11	D N9	A O17	A O20	A + D N19	D N1	D N10	A + D O21	A + D O15	Δd|*RMS* (Å)
TTX	8	N1 D	N15 D	N3 D		O20 A + D		O17 A + D	O18 A + D	O21 A + D	O22 A + D	0.80 1.25
	6	N15 D			O22 A + D	O11 A	O21 A + D	N1 D	O17 A + D			0.40 0.50
	4	N1 D	N15 D		O21 A + D			O22 A + D				0.10 0.10

Matches of acceptor and / or donor heteroatoms. A: Acceptor; D: donor; A + D: acceptor and donor.

ability of the atoms to give or to receive a hydrogen bond. They used Hopfield-like neural networks.[32] They minimized the sum F of the squares of the distance difference matrix elements (Table VI).

We fitted the discrete STX pattern onto the TTX spline approximation, taking account of the ability of the atoms to give and/or to receive hydrogen bonds, according to eq. (10).[10] First, only donor and acceptor heteroatoms were considered; second, all atoms were considered.

Matching acceptor and/or donor atoms only. In STX, only 10 heteroatoms are able to receive or give hydrogen bonds. In TTX, there are 11 such atoms. Thus, the maximal common substructure contains no more than 10 atoms.

TABLE VI.
Tetrodotoxin (TTX) and Saxitoxin (STX) Common Substructure (CSS) Tested by Feuilleaubois et al.[20]

		D	#	#	D	#	D	D	#	A	F	*RMS*
STX	CSS Size	N1	C5	C6	N7	C8	N9	N11	C16	O17	(Å)	(Å)
TTX	9	O22	C6	C13	N1	C2	N3	N15	C12	O21	3.03	1.07
		A + D	#	#	D	#	D	D	#	A + D		

A: acceptor; D: donor; A + D: acceptor and donor; #: neither donor nor acceptor.

TABLE VII.
Tetrodotoxin (TTX) and Saxitoxin (STX) Common Substructures (CSS) Obtained by Fitting the Discrete Structure of STX Onto the Spline Approximation of TTX.

	CSS Size									*RMS* (Å)
STX	4	N1	N7	N11	O17					
		D	D	D	A					0.10
TTX		O22	N1	N15	O21					
		D + A	D	D	D + A					D & D
STX	5	N7	N9	N11	O20	O21				
		D	D	D	A	D + A				0.36
TTX		N1	N3	N15	O20	O11				
		D	D	D	D + A	A				
STX	6	N1	N7	N10	O17	N19	O20			
		D	D	D	A	D + A	A			0.50
TTX		N1	N15	O17	O22	O21	O11			
		D	D	D + A	D + A	D + A	A			D & D
STX	7	N1	N7	N10	O17	N19	O20	O21		
		D	D	D	A	D + A	A	D + A		0.67
TTX		N1	N15	O17	O22	O21	O11	O16		
		D	D	D + A	D + A	D + A	A	D + A		
STX	8	N1	N7	N9	N10	N11	O15	O17	O21	
		D	D	D	D	D	D + A	A	D + A	1.14
TTX		O17	N1	N3	O18	N15	O16	O14	O11	
		D + A	D	D	D + A	D	D + A	A	A	
STX	8	N1	N7	N10	N11	O17	N19	O20	O21	
		D	D	D	D	A	D + A	A	D + A	1.14
TTX		N1	N3	O17	N15	O11	O21	O14	O18	
		D	D	D + A	D	A	D + A	A	D + A	
STX	8	N7	N9	N10	N11	O15	O17	O20	O21	
		D	D	D	D	D + A	A	A	D + A	0.96
TTX		N1	N3	O16	N15	O22	O18	O20	O21	
		D	D	D + A	D	D + A	D + A	D + A	D + A	

Matches limited to heteroatoms able to give and/or receive hydrogen bonds; $\xi \leq 1.5$ Å. A: acceptor; D: donor; A + D: acceptor and donor.

The maximal common substructure size depends on the imposed threshold, ξ. With $\xi = 2$ Å, it is possible to obtain 9 or 10 common atoms. If the threshold is lowered to 1.5 Å, no more than 8 atoms in common are found (Table VII). By comparison of Tables III and VII, it can be seen that our technique provides Danziger and Dean's four-atom and six-atom common substructures. Our best eight-atom common substructure (*RMS* = 0.96 Å) is very similar to the common substructure of the same size found by Danziger and Dean (*RMS* = 1.25 Å) and the fit is better (see Fig. 7, the stereoview of the final superposition). In particular, the nitrogen atoms of the STX guanidinium group (N_7, N_9, and N_{11}) are matched with those of the TTX guanidinium group (N_1, N_3, and N_{15}) and the gem dihydroxyl oxygens (O_{15} and O_{21}) of STX are paired with two cage hydroxyl oxygens of TTX (O_{22} and O_{21}). These nitrogens and hydroxyl oxygens appear to be involved in the binding of the sodium-channel macromolecule.[33,34]

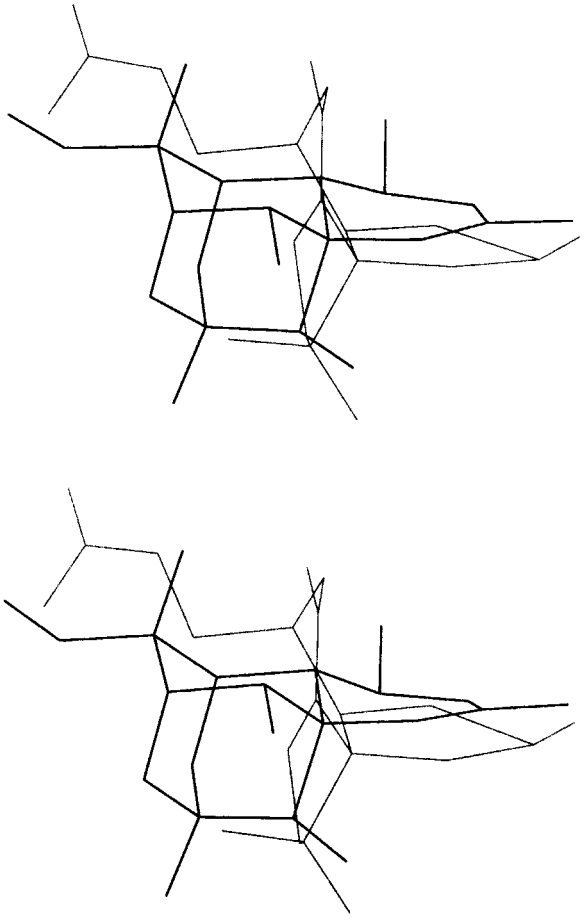


FIGURE 7. Matching eight acceptor and / or donor atoms. Stereoview of the best superposition. Gray line: saxitoxin; Black line: tetrodotoxin.

Matching all atoms. In each structure, 11 atoms are not able to give and/or receive hydrogen bonds. These are the 11 carbon atoms of TTX and the 10 carbon atoms and the N_3 nitrogen of STX. Table VIII shows our best matches. The best 10-atom common substructure is similar to the 8-atom common substructure of Danziger and Dean's study—with there being five identical atoms, the three guanidinium N atoms and two hydroxyl oxygen atoms. It is also similar to the 9-atom common substructure sought by Feuilleaubois et al., in that there are three identical heteroatoms, the three guanidinium nitrogens and two carbon atoms.

In conclusion, our common substructure search compares well with the others. However, we must point out a difficulty inherent in any common substructure search: there are many other possible correspondences of interest that are numerically as good as those discussed or judged (see, e.g., the fifth and sixth matches in Table VIII). Neither of the two criteria, the *RMS* or the local threshold ξ, is sufficient to screen them. Only expert knowledge of the related bioreactivity problem makes it possible to choose.

Conclusion

This article provides a new method for discrete shape similarity recognition. Instead of optimizing in discrete space, our method first optimizes the location of one of the two discrete shapes on a continuous approximation of the other. On the basis of the atom correspondences obtained, the discrete shapes are then overlapped.

Our method requires only few computer resources. It needs, as starting data, the atom Cartesian coordinates and the parameters for the spline approximation of the structures. The storage of this information occupies very little memory space. The central processing time increases roughly logarithmically with the size of the structures to be matched. Consequently, we can match structures with a few hundred atoms.

Current methods for recognizing a given 3D pattern are not able to retrieve patterns common to two structures. We have here applications in the retrieval of a pattern, a nucleotide unit and a sugar, contained many times with some variation in a 316-atom oligonucleotide and in the determi-

TABLE VIII.
Tetrodotoxin (TTX) and Saxitoxin (STX) Common Substructures Obtained by fitting the Discrete Structure of STX onto the Spline Approximation of TTX.

	CSS Size										*RMS* (Å)
STX	4	C2	C4	C6	C16						
		#	#	#	#						0.16
TTX		C7	C5	C13	C12						
		#	#	#	#						
STX	5	C2	N3	C4	C6	N10					
		#	#	#	#	D					0.11
TTX		C12	C13	C6	C8	O21					
		#	#	#	#	D + A					
STX	6	N3	C4	C5	C6	N10	C14				
		#	#	#	#	D	#				0.30
TTX		C5	C10	C12	C13	N3	C9				
		#	#	#	#	D	#				
STX	7	C2	N3	C4	C5	C6	N10	C16			
		#	#	#	#	#	D	#			0.29
TTX		C12	C13	C6	C5	C10	O21	C9			
		#	#	#	#	#	D + A	#			
STX	8	C2	N3	C4	C5	C6	N10	O15	C16		
		#	#	#	#	#	D	D + A	#		0.46
TTX		C12	C13	C6	C5	C10	O21	N1	C9		
		#	#	#	#	#	D + A	D	#		
STX	9	C2	C4	C5	C6	N7	C13	C16	O20	O21	
		#	#	#	#	D	#	#	A	D + A	0.36
TTX		C12	C8	C7	C6	O17	C19	C5	O16	O18	
		#	#	#	#	D + A	#	#	D + A	D + A	
STX	10	C4	C5	C6	N7	C8	N9	N11	C16	O20 O21	
		#	#	#	D	#	D	D	#	A D + A	0.39
TTX		C5	C6	C7	N1	C2	N3	N15	C8	O20 O11	
		#	#	#	D	#	D	D	#	D + A A	

Matches of all atoms; $\xi \leq 1$ Å. A: acceptor; D: donor; A + D: acceptor and donor; #: neither donor nor acceptor.

nation of common patterns for two marine neurotoxins. Our results compare well with those of other investigators.

Most of the current methods for shape similarity analysis minimize a distance difference matrix and achieve a good local match at the expense of an overall fit. For these two reasons, they are not able to measure the dissimilarity of two enantiomers of a chiral body. Our method, which works on Cartesian coordinates and searches for an overall match, might be able to achieve this. We shall examine this point in a subsequent study.

References

1. P. Ehrlich and J. Morgenroth, *On Haemolysis: Third Communication. The Collected Papers of Paul Ehrlich, Vol. 1*, F. Himmelweit, Ed., Pergamon Press, London, 1956, p. 205; L. B. Kier, *Molecular Orbital Theory in Drug Research*, Academic Press, New York, 1971.
2. F. H. Allen, O. Kennard, and R. Taylor, *Acc. Chem. Res.*, **16**, 146 (1983).
3. E. E. Abola, F. C. Bernstein, and T. F. Koetzle, *The Protein Data Bank in the Role of Data in Scientific Progress*, P. S. Glaeser, Ed., Elsevier, New York, 1985.
4. R. S. Pearlman, *Concord User's Manual*, Tripos Associates, St. Louis, MO, 1987; R. S. Pearlman, *Chem. Des. Automat. News*, **2**, 1 (1987).
5. C. Hansch and T. Fujita, *J. Am. Chem. Soc.*, **86**, 1616 (1964).
6. L. Hammett, *Physical Organic Chemistry*, McGraw-Hill, New York, 1970.
7. P. Gund, W. T. Wipke, and R. Langridge, *Proc. Int. Conf. Comput. Chem. Res. Edu.*, Ljubljana, 1973, p. 33; P. Gund, W. T. Wipke, and R. Langridge, *Comput. Chem. Res. Edu. Technol.*, **3**, 5 (1974).
8. N. L. Allinger, *Pharmacology and the Future of Man, Vol. 5, Proceedings of the Fifth International Congress*, R. A. Maxwell, Ed., Karger, Basel, 1972, p. 57.

9. A. Y. Meyer and W. G. Richards, *J. Comput.-Aid. Mol. Des.*, **5**, 427 (1991).
10. M. Petitjean, *J. Comput. Chem.*, **16**, 80 (1995).
11. R. Carbo, L. Leyda, and M. Arnau, *Int. J. Quant. Chem.*, **17**, 1185 (1980); R. Carbo and L. Domingo, *Int. J. Quant. Chem.*, **32**, 517 (1987).
12. E. E. Hodgkin and W. G. Richards, *Int. J. Quant. Chem. Quant. Biol. Symp.*, **14**, 105 (1987); C. Burt, W. G. Richards, and P. Huxley, *J. Comput. Chem.*, **11**, 1139 (1990); A. C. Good, E. E. Hodgkin, and W. G. Richards, *J. Chem. Inf. Comput. Sci.*, **32**, 188 (1992).
13. A. M. Lesk, *Commun. ACM*, **22**, 219 (1974).
14. S. E. Jakes, N. Watts, P. Willett, D. Bawden, and J. D. Fischer, *J. Mol. Graph.*, **5**, 41 (1987).
15. V. Golender and A. Rozenblit, *Logical and Combinatorial Algorithms for Drug Design*, Research Studies Press, Letchworth, UK, 1983.
16. F. S. Kuhl, G. M. Crippen, and D. K. Friesen, *J. Comput. Chem.*, **5**, 24 (1984).
17. A. T. Brint and P. Willett, *J. Mol. Graph.*, **5**, 49 (1987).
18. J. R. Ullman, *J. ACM*, **16**, 31 (1976).
19. D. J. Danziger and P. M. Dean, *J. Theor. Biol.*, **116**, 215 (1985).
20. E. Feuilleaubois, V. Fabart, and J. P. Doucet, *SAR and QSAR in Environ. Res.*, **1**, 97 (1993).
21. M. T. Barakat and P. M. Dean, *J. Comput.-Aided Mol. Des.* **4**, 295 (1990).
22. P. G. Ciarlet, *Introduction à l'analyse numérique matricielle et à l'optimisation*, Masson, Paris, 1994.
23. J. Broyden, *J. Inst. Math. Appl.*, **6**, 222 (1970); R. Fletcher, *Comput. J.*, **13**, 317 (1970); B. Goldfarb, *Math. Comp.*, **24**, 23 (1970); D. F. Shanno, *Math. Comp.*, **24**, 647 (1970).
24. Y. Lin, M. Risk, S. M. Ray, D. Van Engen, J. Clardy, J. Golik, J. C. James, and K. Nakanishi, *J. Am. Chem. Soc.*, **103**, 6773 (1981).
25. G. Dodin and C. Cordier, personal communication.
26. R. Lavery, H. Sklenar, K. Zakrzewska, and B. Pullman, *J. Biomol. J. Struct. Dynam.*, **3**, 989 (1986); R. Lavery, *Structure and Expression, Vol. 3*, W. K. Olson, R. H. Sarma, M. H. Sarma, and M. S. Sundaraligam, Eds., Adenine Press, New York, 1988, p. 191.
27. A. Rassat, *C. R. Acad. Sci., Paris*, **299** (Série II), 53 (1984).
28. F. Hausdorff, *Set Theory, 2nd Ed.*, Chelsea Publishing, New York, 1962, p. 166.
29. G. Dodin, J. M. Kühnel, P. Demerseman, and J. Kotzyba, *Anti-Cancer Drug Des.*, **8**, 416 (1993); G. Dodin, B. Bourliataud, C. Cordier, and J. P. Blais, *J. Org. Chem.*, **61**, 2561 (1996).
30. M. Poncin, B. Hartmann, and R. Lavery, *J. Mol. Biol.*, **226**, 775 (1992).
31. P. M. Dean and P. L. Chau, *J. Mol. Graph.*, **5**, 152 (1987).
32. J. J. Hopfield, *Biol. Cybernet.* **52**, 141 (1985); J. J. Hopfield, *Proc. Natl. Acad. Sci. USA*, **81**, 3088 (1984).
33. B. Hille, *J. Biophys.* **15**, 615 (1975).
34. C. Kao and S. E. Walker, *J. Physiol.* (Lond), **323**, 619 (1982).